

# Cálculo de valores propios de una matriz

Como viste en la clase anterior, cuando tenemos un conjunto de datos con muchas variables, la matriz de covarianza es el objeto matemático que permite identificar cuáles variables están altamente correlacionadas.

Cuando dos variables están altamente correlacionadas, quiere decir que aportan básicamente la misma información del problema y eso podría indicar que solo necesitaríamos una sola de ellas. En la próxima clase veremos que esa reducción de variables se puede hacer con una técnica matemática denominada PCA (Principal Component Analysis).

Esta técnica está basada en un concepto del álgebra de vectores y matrices, que llamamos el cálculo de los valores propios de una matriz, y en esta lectura profundizaremos sobre qué significa ese procedimiento.

## Repaso de matrices

Las matrices en general son objetos matemáticos que tienen un cierto número de filas y columnas (estos números los denominamos dimensiones de la matriz).

Las matrices tienen una operación especial, que llamamos transponer, la cual consiste en ubicar cada fila como una columna en una nueva matriz, el resultado de una transposición se denomina matriz transpuesta y se ve así:

$$\begin{bmatrix} 4 & -3 \\ 1 & 2 \end{bmatrix}^T = \begin{bmatrix} 4 & 1 \\ -3 & 2 \end{bmatrix}$$

Así también, entre las matrices podemos definir reglas de suma y resta de forma similar a como hacemos con los números naturales o decimales, con una condición especial: **ambas matrices deben tener las mismas dimensiones** y cuando las dimensiones de una matriz son **iguales entre ellas** ( $\# \text{ filas} = \# \text{ columnas}$ ) decimos que la matriz es **cuadrada**.

Con esto en mente resumimos lo anterior con un ejemplo de suma entre dos matrices cuadradas así:

## Suma

$$\begin{bmatrix} 1 & 3 \\ 2 & 0 \end{bmatrix} + \begin{bmatrix} -1 & 1 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} 1 + (-1) & 3 + 1 \\ 2 + 3 & 0 + 2 \end{bmatrix} = \begin{bmatrix} 0 & 4 \\ 5 & 2 \end{bmatrix}$$

Donde vemos que la suma de matrices se hace elemento a elemento para dar origen a otra matriz con las mismas dimensiones. Entonces para obtener el elemento de la primera fila y primera columna de la matriz suma, se suman los elementos correspondientes a cada matriz ubicados en la primera fila y primera columna:

$$1 + (-1) = 0$$

Ahora bien, las matrices también tienen una operación de multiplicación entre ellas que es más compleja de definir que la suma, sin embargo aquí vamos a desglosarla.

## Multiplicación

Vamos a reemplazar los elementos numéricos de las matrices del ejemplo anterior por variables algebraicas, indicando que pueden ser cualquier número para así exponer el proceso con la mayor generalidad posible. De esta manera, vamos a definir el producto de dos matrices cuadradas de tal manera que el resultado sea otra matriz de las mismas dimensiones así:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

Donde cada expresión algebraica tiene la forma de dos índices que denotan la fila y columna donde está posicionado el número, respectivamente.

La manera de calcular cada elemento de la matriz resultante viene dado por una regla sencilla que ilustraremos con el siguiente ejemplo:

$$\begin{matrix} & & \underbrace{a_{11}b_{11}}_{\text{red}} + \underbrace{a_{12}b_{21}}_{\text{blue}} = c_{11} \\ & \nearrow & \nwarrow \\ \begin{bmatrix} \boxed{a_{11}} & \boxed{a_{12}} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} \boxed{b_{11}} & b_{12} \\ \boxed{b_{21}} & b_{22} \end{bmatrix} = \begin{bmatrix} \boxed{c_{11}} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \end{matrix}$$

Que equivale a decir que el resultado del elemento de la fila 1 y columna 1 de la matriz se calcula como la suma de los productos de los elementos de la primera fila de la primera matriz por los elementos de la primera columna de la segunda matriz

$$c_{11} = a_{11}b_{11} + a_{12}b_{21}$$

Veamos otro ejemplo para tenerlo más claro:

$$\begin{matrix} & & \underbrace{a_{11}b_{12}}_{\text{red}} + \underbrace{a_{12}b_{22}}_{\text{blue}} = c_{12} \\ & \nearrow & \nwarrow \\ \begin{bmatrix} \boxed{a_{11}} & \boxed{a_{12}} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & \boxed{b_{12}} \\ b_{21} & \boxed{b_{22}} \end{bmatrix} = \begin{bmatrix} c_{11} & \boxed{c_{12}} \\ c_{21} & c_{22} \end{bmatrix} \end{matrix}$$

En este caso el resultado del elemento de la fila 1 y columna 2 de la matriz se calcula como la suma de los productos de los elementos de la primera fila de la primera matriz por los elementos de la segunda columna de la segunda matriz.

## Multiplicación en Python

En python el producto de matrices se calcula fácil usando la librería numpy:

```
np.matmul(A, B)
```

Donde, por ejemplo, una matriz:

$$\begin{bmatrix} 2 & 4 \\ -1 & 2 \end{bmatrix}$$

Se escribe en python como:

```
A = np.array([[2, 4], [-1, 2]])
```

Esta definición se hizo para matrices de dos filas y dos columnas que denominamos matrices 2x2. Pero se aplica de la misma manera para matrices cuadradas de cualquier tamaño NxN. Por ahora solo nos interesará esta definición restringida para matrices cuadradas, pero en cualquier curso de álgebra de matrices podrás darte cuenta de que esta definición se puede hacer más general para matrices que no sean necesariamente cuadradas.

## Matriz Inversa

Ahora, así como cada número tiene su inverso, donde el inverso se define como aquel número tal que la multiplicación de ambos da 1:

$$6 \cdot \frac{1}{6} = 1$$

En este caso  $\frac{1}{6}$  es el inverso de 6. Así también, las matrices pueden tener su inversa (aunque no siempre), la matriz inversa  $A^{-1}$  de una matriz dada  $A$  se define como aquella matriz que cumpla la siguiente igualdad:

$$A \cdot A^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

## Matriz Inversa en Python

El cálculo de matriz inversa se hace rápido usando NumPy nuevamente así:

```
A = np.array([[2, 4], [-1, 2]])
Ainversa = np.linalg.inv(A)
```

Donde, el resultado de A inversa en este caso particular sería:

```
array([[ 0.25 , -0.5  ],
       [ 0.125,  0.25 ]])
```

Para comprobar que esto está bien, puedes multiplicar ambas matrices para ver que da lo correcto:

```
np.matmul(A, Ainversa)
```

```
In [ ]: import numpy as np
A=np.array([[2,4],[-1,2]])
A_inv=np.linalg.inv(A)
print(f'A = {A}\nA inversa = {A_inv}\nA*A inversa = {np.matmul(A,A_inv)}')
```

```
A = [[ 2  4]
      [-1  2]]
A inversa = [[ 0.25 -0.5 ]
             [ 0.125 0.25 ]]
A*A inversa = [[1. 0.]
               [0. 1.]]
```

## Repaso de vectores

Como caso particular adicional a la definición anterior, consideremos el producto de una matriz cuadrada por un vector (aquí entendemos un vector como una matriz de una sola columna, también se le denomina por eso vectores columna en muchos libros de álgebra lineal) cuya longitud es igual al número de filas de la matriz así:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = \begin{bmatrix} c_{11} \\ c_{21} \end{bmatrix}$$

Donde definimos que el producto de una matriz por un vector resulta en otro vector de las mismas dimensiones (filas). Y la regla que consideramos para el caso de matrices cuadradas también aplica de manera que los elementos del vector resultante se obtendrían de multiplicar filas por columnas así:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = \begin{bmatrix} c_{11} \\ c_{21} \end{bmatrix}$$

También tenemos una operación entre vectores que denominamos el producto punto o producto interior. Normalmente al considerar esta definición se representa el primer vector como una sola fila y el segundo como una sola columna así (sigamos pensando con base en el ejemplo anterior de la matriz por el vector, pero ahora la matriz solo tiene una fila):

$$\begin{bmatrix} a_{11} & a_{12} \end{bmatrix} \cdot \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = \begin{bmatrix} c_{11} \end{bmatrix}$$

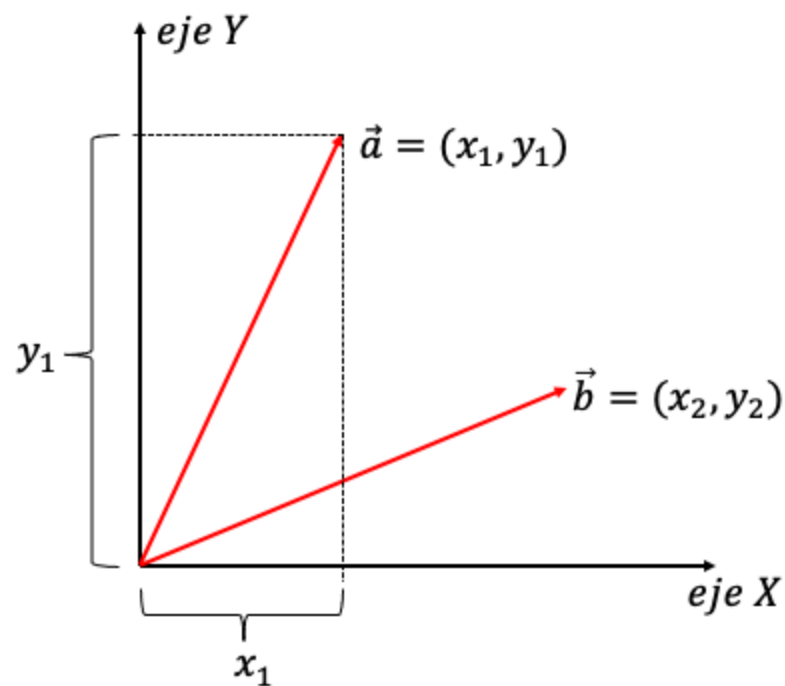
Y como ya te estás dando cuenta (teniendo en mente la misma definición de multiplicación de matrices) al solo haber una fila en la primera matriz y una columna en la segunda matriz, el resultado solo podrá tener un elemento y es por esto que el resultado de multiplicar dos vectores de esta manera es siempre un número:

$$\vec{a} \cdot \vec{b} = \begin{bmatrix} a_{11} & a_{12} \end{bmatrix} \cdot \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = \begin{bmatrix} c_{11} \end{bmatrix} = c_{11} = a_{11}v_{11} + a_{12}v_{21}$$

Simplifiquemos la notación así:

$$\vec{a} \cdot \vec{b} = \begin{bmatrix} x_1 & y_1 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = x_1x_2 + y_1y_2$$

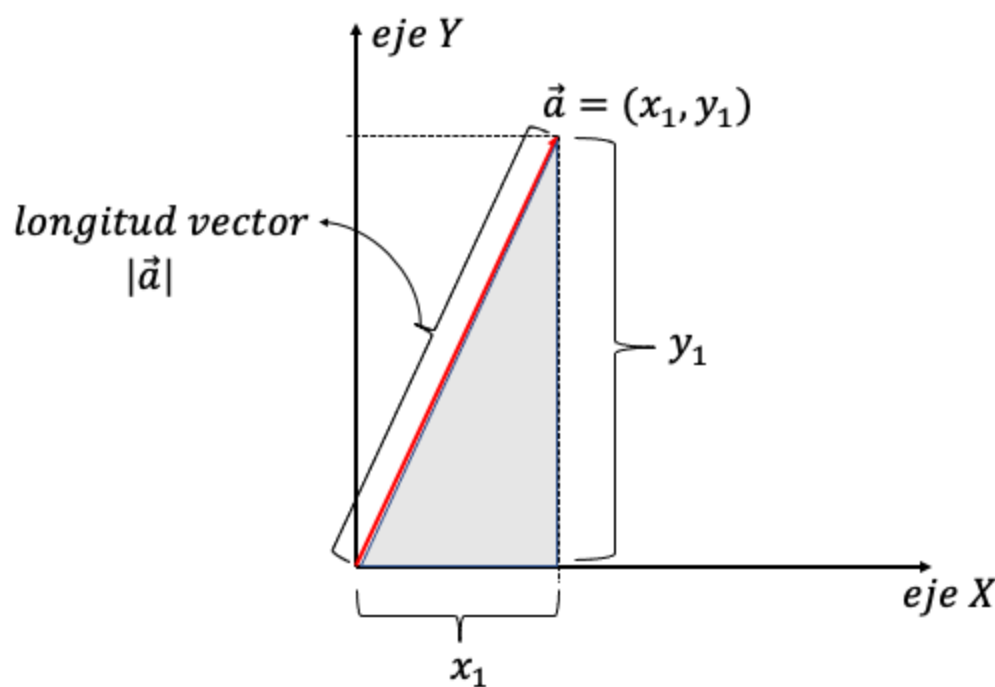
Y esto nos recuerda la clásica regla de multiplicar vectores como x por x más y por y. Y si los vectores tienen más dimensiones, entonces ... más z por z y así. Recuerda que la notación simplificada es porque ahora tenemos que la primera componente es el eje X del vector y la segunda componente el eje Y del vector cuando lo dibujamos en un plano cartesiano.



Ahora, cuando pensamos en multiplicar un vector por el mismo, la misma definición aplica:

$$\vec{a} \cdot \vec{a} = \begin{bmatrix} x_1 & y_1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = x_1^2 + y_1^2$$

Y nos damos cuenta de que esto se relaciona con el Teorema de Pitágoras al ver que el producto de un vector por el mismo nos da el cuadrado de la longitud de la flecha que representa al vector en el plano cartesiano:



$$x_1^2 + y_1^2 = |\vec{a}|^2$$

Así vemos que la longitud de un vector, también conocida como norma del vector se calcula como:

$$|\vec{a}| = \sqrt{x_1^2 + y_1^2} = \sqrt{\vec{a} \cdot \vec{a}}$$

Listo, con esto terminamos un repaso básico de lo mínimo de matrices y vectores para lo que viene.

## Vectores y Valores propios de una matriz

En álgebra lineal podemos tener ecuaciones donde la incógnita es un vector, supongamos la siguiente ecuación:

$$A \cdot \vec{x} = \lambda \vec{x}$$

Aquí A es una matriz cuadrada NxN cuyos elementos conocemos perfectamente y X es un vector columna cuyas componentes desconocemos. Aquí recordemos que multiplicar un vector por un número es simplemente multiplicar cada componente del vector por dicho número.

Entonces lo que esta ecuación nos pregunta es:

¿Existen vectores X tales que al multiplicarlos por la matriz A eso es equivalente a simplemente multiplicarlos por un número?

Si tal vector existe y está asociado a un valor específico de  $\lambda$ , entonces decimos que el vector X es un vector propio de la matriz A y  $\lambda$  es su valor propio correspondiente.

Consideremos esto para el caso de una matriz 2 x 2, como la siguiente:

$$\begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} x \\ y \end{bmatrix}$$

Esto se traduce en el sistema de ecuaciones (haciendo el producto matriz por vector):

$$\begin{bmatrix} x + 2y \\ x \end{bmatrix} = \begin{bmatrix} \lambda x \\ \lambda y \end{bmatrix}$$

Aquí entonces debemos encontrar las combinaciones de x e y, que satisfacen el sistema de ecuaciones. En general, hacer esto requiere otros conceptos más detallados del álgebra de matrices como el cálculo de determinantes y resolver ecuaciones polinomiales cuya explicación solo puede dejarse a un curso exclusivo de álgebra lineal. Pero no te preocupes, ya que podemos hacer este cálculo de manera rápida con python así:

```
import numpy as np

A = np.array([[1, 2], [1, 0]])
values, vectors = np.linalg.eig(A)
Donde la matriz A contiene los elementos exactos de la matriz anterior y el comando np.linalg.eig(A) lo que hace es calcular directamente los valores y vectores propios, llamados values y vectors en el código, respectivamente.
```

Verás que esta matriz tiene dos valores propios:

```
array([ 2., -1.])
Con sus respectivos vectores propios asociados:
```

```
array([[ 0.89442719, -0.70710678 ],
       [ 0.4472136 , 0.70710678 ]])
Aquí es importante anotar que los vectores que entrega la función np.linalg.eig(A) son vectores columna de manera que los elementos de la primera columna de vectors corresponden con el primer valor de values y así sucesivamente. Entonces en nuestro lenguaje matemático usual, escribimos las dos soluciones como:
```

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.89442719 \\ 0.4472136 \end{bmatrix}, \lambda = 2$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -0.70710678 \\ 0.70710678 \end{bmatrix}, \lambda = -1$$

Puedes verificar que cada vector y su respectivo valor propio cumplen la ecuación original ejecutando cada parte así:

```
np.matmul(A, vectors.T[1])
```

Que te da como resultado:

```
array([ 0.70710678, -0.70710678])
```

Mientras que por otro lado calculando:

```
values[1]*vectors.T[1]
```

Resulta en lo mismo:

```
array([ 0.70710678, -0.70710678])
```

Donde hemos considerado el segundo vector y valor propio respectivamente tomando  $\lambda = -1$  y el vector incógnita X igual a `vectors.T[1]`.

## Diagonalizacion de Matriz

Uno de los hechos más importantes de obtener los vectores y valores propios de una matriz es poder diagonalizarla. En general se define que una matriz A es diagonalizable si es posible escribirla como el producto de:

$$A = P \cdot D \cdot P^{-1}$$

Donde D es una matriz diagonal (matriz donde todos los elementos por fuera de la diagonal son cero), un ejemplo de matriz diagonal sería:

$$D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

Y aquí un resultado matemático bien conocido es que si una matriz es diagonalizable, la matriz D se construye colocando sus valores propios en la diagonal y la matriz P se construye colocando en cada columna el vector propio, siguiendo el mismo orden de valores propios correspondientes de la matriz D, así:

$$P \cdot D \cdot P^{-1} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \vec{x_1} & \vec{x_2} & \vec{x_3} \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ \vec{x_1} & \vec{x_2} & \vec{x_3} \\ \vdots & \vdots & \vdots \end{bmatrix}^{-1}$$

Lo importante de estudiar este procedimiento en nuestro curso, es que cuando aplicamos este cálculo de vectores y valores propios a una matriz de covarianza, los vectores representan las direcciones a lo largo de las cuales percibimos la mayor cantidad de varianza de ese conjunto de datos, donde la cantidad de varianza es proporcional al valor propio de cada vector propio.

Y es importante tener en cuenta que este procedimiento aplica para un conjunto de datos con N variables al que le corresponde una matriz de covarianza de tamaño NxN.

Ahora, el último factor importante de esta técnica es que para matrices de covarianza, sus vectores propios siempre son independientes unos de otros y esto es justamente lo que queremos en un proceso de reducción de variables, porque direcciones independientes implica que estos vectores representan nuevas variables cuya correlación es la más baja posible y así cada nueva variable es lo más representativa posible.

En álgebra lineal se dice más precisamente que los vectores propios de una matriz de covarianza son ortogonales y esto quiere decir que el producto interno de cualquier par de estos vectores siempre da como resultado cero:

$$\vec{x_i} \cdot \vec{x_j} = 0$$

Como consecuencia la matriz se denomina matriz ortogonal, y se sabe en matemáticas que la inversa de una matriz ortogonal es igual a la transpuesta, de manera que:

$$A = P \cdot D \cdot P^{-1} = P \cdot D \cdot P^T$$

En la próxima clase veremos esto de forma más detallada y visual, te veo allí.

### Extras:

- [Diagonalizacion de Matriz](#)
- [Definición de valor y vector propio](#)
- [Playlist algebra lineal](#)
- [Eigen/07%3A\\_Spectral\\_Theory/7.01%3A\\_Eigenvalues\\_and\\_Eigenvectors\\_of\\_a\\_Matrix\)](#)